

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (currently amended) A computer-readable storage medium encoded with computer-executable instructions to perform a method of validating ~~a message~~ text content structured into a plurality of elements comprised of markup and the content, which is delimited by the mark, the method comprising:

encountering a first element of the ~~message~~ plurality of elements;

consulting ~~a first~~ one or more validation tables to identify, from among a plurality of validation delegates, a first delegate that corresponds to said first element, the one or more validation tables not being schema that describe valid syntax or structure of said text content;

~~applying~~ executing said first delegate ~~to said~~ with the content of the first element to determine whether the content in the first element is valid;

encountering a second element that is a root of a subtree of said first element;
~~and~~

consulting the one or more validation tables to identify, from among the plurality of validation delegates, a second delegate; and

~~applying a~~ executing said second delegate to with the content of the second element to determine whether the content in the second element is valid ~~said second element in accordance with said first validation table~~,

wherein at least one of said first delegate ~~and~~ or said second delegate determines whether the content of the element to which it is applied is valid based on at least one non-syntactic condition of the corresponding content ~~whose truth is not assessable solely from the message's syntax~~.

2. (currently amended) The computer-readable storage medium of claim 1, further comprising:

prior to consulting said ~~first validation table~~ one or more validation tables, determining that ~~a second validation table~~ at least one of the one or more validation tables contains no delegate that corresponds to said first element.

3. (currently amended) The computer-readable storage medium of claim 1, wherein said act of ~~applying~~ executing said first delegate is performed prior to said act of encountering said second element, and wherein the method further comprises:

subsequent to ~~applying~~ executing said second delegate, consulting said first validation table to identify a third delegate that corresponds to said first element; and

~~applying~~ executing said third delegate ~~to~~ with the content of said first element.

4. (currently amended) The computer-readable storage medium of claim 1, wherein said ~~first validation table includes~~ one or more validation tables include, for each element listed in said ~~first validation table~~ one or more validation tables, a flag indicating whether subtrees of a given element are to be traversed, and wherein the method further comprises:

determining that the flag corresponding to said first element indicates that subtrees of said first element are to be traversed.

5. (currently amended) The computer-readable storage medium of claim 1, wherein the method further comprises:

selecting, according to a criterion, ~~said a~~ first validation table from among a ~~plurality of validation tables~~ said one or more validation tables, the ~~selected first~~ validation table being the table that is consulted to identify delegates to be applied.

6. (currently amended) The computer-readable storage medium of claim 1, wherein said first delegate makes at least one decision based on an element that is nether said first element nor a subtree of said first element.

7. (currently amended) The computer-readable storage medium of claim 1, wherein said ~~first validation table includes~~ one or more validation tables include, for each element listed in said ~~first validation table~~ one or more validation tables, a flag indicating whether subtrees of a given element are to be traversed, and wherein the method further comprises:

determining that the flag corresponding to said second element indicates that subtrees of said second element are not to be traversed; and

applying said second delegate to said second element and to a subtree rooted at said second element.

8. (currently amended) The computer-readable storage medium of claim 1, wherein said first delegate and said second delegate each comprise interpretable code.

9. (currently amended) The computer-readable storage medium of claim 1, wherein the method further comprises:

determining whether the message text content is valid based on ~~application execution~~ of delegates ~~to the message with the text content~~ in accordance with said one or more validation tables ~~including said first validation table~~.

10. (currently amended) A method of validating ~~of a message~~ text content structured into a plurality of elements comprised of markup and the content, which is delimited by the markup, comprising:

creating one or more validation delegates, each validation delegate being adapted to ~~validate~~ determine whether a particular type of element of the ~~message text is~~ valid;

creating a first validation table that specifies at least one of the validation delegates for each type of element of the ~~message text~~; and

applying a validation engine to the ~~message text~~, wherein the validation engine, for each of the elements ~~of the message~~, consults said first validation table to identify, from among the validation delegates specified by the table, a validation delegate to be ~~applied~~ to executed with the element, and ~~applies~~ executes the identified validation delegate ~~to with~~ the content of the element,

wherein at least one of the validation delegates determines whether the content of the element ~~to with~~ which it is ~~applied~~ executed is valid based on at least one non-syntactic condition of the content of the element ~~whose truth is not assessable solely from the message's syntax~~.

11. (currently amended) The method of claim 10, further comprising:
creating a second validation table that specifies a validation delegate for each type of element of the ~~message~~ text content,
wherein the validation engine consults the second validation table and, if there is an entry for a given type of element, ~~applies~~ executes the validation delegate specified by the second validation table, and otherwise ~~applies~~ executes the validation delegate specified by the first validation table.

12. (currently amended) The method of claim 10, wherein the first validation table specifies first and second validation delegates for each type of element of the ~~message~~ text content, and wherein the validation engine ~~applies~~ executes the first validation delegate to the element when the element is encountered, and ~~applies~~ executes the second validation delegate to the element after the first validation delegate, and the validation delegates applicable to any subtrees of the element, have been ~~applied~~ executed.

13. (currently amended) The method of claim 10, wherein the first validation table specifies, for a first one of the elements of the ~~message~~ text content, that a subtree of said first one of the elements are not to be traversed, and wherein the method further comprises ~~applying the validation delegate identified for said first one of the elements by said first validation table to all of the elements in a subtree rooted at said first one of the elements without said validation engine traversing elements in the subtree rooted at said first one of the elements~~ giving the validation delegate identified for said first one of the elements by said first validation table exclusive control over said first one of the elements and any subtree of said first one of the elements.

14. (currently amended) The method of claim 10, wherein said ~~message~~ text content is organized in the form of a tree, each element of the ~~message~~ text content corresponding to a node in the tree.

15. (currently amended) The method of claim 10, wherein said first validation table specifies, for each type of element in the ~~message~~ text content, a flag indicating whether the validation delegates for subtrees of said type of element are to be applied to said subtrees.

16. (original) The method of claim 10, wherein each of said validation delegates comprises interpretable code.

17. (original) The method of claim 10, wherein at least one of the validation delegates comprises a null delegate.

18. (currently amended) A computer-readable storage medium having stored thereon a data structure comprising:

a plurality of element names, each element name identifying ~~a portion of a message~~ an element within text content that is structured into a plurality of elements comprised of markup and the content, which is delimited by the markup; and

for each element name, a corresponding name of a first delegate, each first delegate comprising code that ~~validates~~ determines whether the element ~~of the message~~ that corresponds to the element name is valid, wherein the first delegate determines whether the element that it is called upon to validate is valid based on at least one non-syntactic condition ~~whose truth is not assessable solely from the message's syntax~~ of the content that corresponds to that element.

19. (currently amended) The computer-readable storage medium of claim 18, wherein the data structure further comprises:

for each element name, a corresponding name of a second delegate, each second delegate comprising code that executes after the first delegate corresponding to the element name, and any delegates corresponding to a subtree of the element that corresponds to the element name, have been run.

20. (currently amended) The computer-readable storage medium of claim 18, wherein the data structure further comprises:

for each element name, a flag indicating whether the first delegate corresponding to the element name is to be applied exclusively to the element corresponding to the element name and to any subtree thereof.

21. (currently amended) A system for validating ~~a message~~ text content that comprises a plurality of elements that comprise markup and content, the text content being organized in the form of a tree, the system comprising:

a first validation table indicating a plurality of delegates, each delegate corresponding to a name of an element in the tree;

a validation engine that traverses the tree and, for each element encountered in the traversal of the tree, consults the first validation table to identify one of the plurality of delegates that is to be ~~applied to~~ executed with the encountered element, and that ~~applies~~ executes the identified delegate by invoking the delegate, wherein at least one of the plurality of delegates determines whether the element ~~to~~ with which it is ~~applied~~ executed is valid based on at least one non-syntactic condition ~~whose truth is not assessable solely from the message's syntax~~.

22. (original) The system of claim 21, further comprising:

a second validation table indicating a plurality of delegates, each delegate corresponding to a name of an element in the tree; wherein the validation engine, for each element encountered in the traversal of the tree, consults the second validation table to determine whether there is an entry for the encountered element, the validation engine using the entry for the encountered element in the second validation table if there is such an entry, the validation engine consulting the first validation table if the second validation table contains no entry corresponding to the encountered element.

23. (original) The system of claim 21, wherein the first validation table identifies a post-handling delegate for at least some of the elements in the tree, the post-handling delegate being invoked after the delegate for the element, and the delegates for the element's subtrees, have been invoked.

24. (currently amended) The system of claim 21, wherein the first validation table indicates, for a first one of the encountered elements, that the delegate identified by the first validation table for said first one of the encountered elements is to be applied to a subtree rooted at said first one of the encountered elements, and wherein ~~the validation engine applies the delegate identified by the first validation table for said first one of the encountered elements to said subtree without said validation engine traversing said subtree~~ the validation delegate identified for said first one of the elements by said first validation table is given exclusive control over said first one of the elements and any subtree of said first one of the elements.

25. (original) The system of claim 21, wherein each of the delegates comprises interpretable code.

26. (new) A computer readable storage medium storing a validation engine that performs a process, the process performed by the validation engine comprising:

traversing a structured document to visit elements of the structured document, the elements comprising markup and content that is delimited by the markup, where the markup conforms to a markup language, and where the elements are of various element types; and

checking the types of the respective elements when the validation engine visits the elements for validation and using the types to determine which executable validation delegates to execute to validate the contents of the respective elements, where the determining is performed according to mapping information that maps the executable validation delegates to corresponding element types, and where the mapping information can be modified to cause the same validation engine, without modification, to modify which executable validation delegates the validation engine selects for which corresponding element types.

27. (new) A computer readable storage medium storing a validation engine according to claim 26, wherein the executable validation delegates validate that the content satisfies non-syntactic conditions.

DOCKET NO.: MSFT-1956/303857.1
Application No.: 10/643,031
Office Action Dated: September 8, 2006

**PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116**

28. (new) A computer readable storage medium storing a validation engine according to claim 26, wherein the traversing is recursive and the content of at least one element is determined to be valid or not based on a delegate selected for the at least one element and based on another delegate selected for an element in a sub-tree of the at least one element.